

**PROVISIONAL PATENT APPLICATION**

**MULTIPLE OBJECT DOWNLOAD**

Inventor(s): Nathan F. Raciborski, a citizen of the United States, residing at,  
2643 East Spring Rd.  
Phoenix, AZ 85032

Mark R. Thompson, a citizen of the United States, residing at,  
1130 West Longhorn Drive  
Chandler, AZ 85248

Assignee: Limelight Networks, LLC  
8936 North Central Avenue  
Phoenix, AZ 85020

Entity: Small entity

## **MULTIPLE OBJECT DOWNLOAD**

[01] The ensuing description provides preferred exemplary embodiment(s) only, and is not intended to limit the scope, applicability or configuration of the invention. Rather, the ensuing description of the preferred exemplary embodiment(s) will provide those skilled in the art with an enabling description for implementing a preferred exemplary embodiment of the invention. It being understood that various changes may be made in the function and arrangement of elements without departing from the spirit and scope of the invention as set forth in the appended claims.

[02] This invention solves a problem that doesn't seem to have a better solution today.

More specifically, there is no way to easily download multiple content objects. An example of the problem is demonstrated at a web site located at buymusic.com that is forcing people to individually click on each object to save the object to their local drive. For a complete album that could be 14 or more individual downloads for each song, for example.

[03] Other attempted solutions to this problem are generally limited to creating a client application for downloading groups of files. The client application, which is not tailored for each group of files, is initiated during a download of a file having a group of objects by prompting the operating system for the application with a unique file extension associated in the operating system with the client application. This method requires defining a unique file extension to the operating system so the appropriate client application is retrieved. This becomes cumbersome when trying to get users to install and upgrade the client application. To download the files on another computer, both the application and another file is required to tell the application which files to download.

[04] Another technique for solving the problem remotely creates a single compressed file that includes the media objects, for example, a zip compressed file with multiple objects. The zip file is very large and ultimately has to be unzipped by the user, which can be problematic.

[05] In one embodiment, the present invention provides the ability to download multiple objects at one time in an improved manner. The invention dynamically creates a content group program unique to the user, transaction session identifier and/or the group of objects. Execution of the content group program downloads a specific list of objects associated with the transaction session identifier. As part of the download process, the content group program identifies the user and/or purchase to the origin server and identifies the objects to download. A certificate or token could be embedded in the content group program to allow

authentication of the content group program by providing the certificate to the origin server. The user can pass the dynamically and specifically created program to another account or computer and re-execute it. On the other computer, the program will initiate the downloads to the present computer.

5    **[06]**    In another embodiment, the content group program includes a transaction session identifier. The origin server or related server can retrieve other information using the unique transaction session identifier. For example, a list of objects to download and a user identifier are retrieved by the origin server using the transaction session identifier. Once the list of objects are known and the download is authorized, the download of the objects is initiated in  
10   a serial or parallel fashion.

**[07]**    The content group program is an executable program that initiates the download of specified objects from a network when initiated. In this embodiment, the content group program is about 150KB in size, but could be other sizes in other embodiments. There could be various versions of the content group program to allow execution on various software  
15   platforms, or a version could be cross-platform compatible to execute on a number of software platforms. Each content group program is unique to a particular transaction session. Once the user selects the objects for download, the content program group is generated for that transaction session. A button on the web page corresponds to the content group program. A right-click on the button, for example, initiates download of the content group program to a  
20   specified location using the HTTP download function of the web browser. The user can run the content group program immediately after download or at any later time.

**[08]**    Execution of the content group program causes download of the objects to any path specified by the user. The content group program has an interface that: shows the objects being download, allows selection of a subset of the objects in the predefined group, allows  
25   interruption of the download of all or some objects, allows adding additional objects to the download, provides status information on the downloads, provides usage information for any DRM of the objects, etc. Only the single content program is required to initiate download of the objects on any computer. This allows sending this file in e-mail to any recipient. The recipient doesn't need to install the content group program, but only has to execute the  
30   program to download the objects to any specified path.

**[09]**    Generally, the content group program has two components in this embodiment, more specifically, a program portion and a payload portion. The program portion is generic to any payload portion. In this embodiment, the program portion is tamper proof so as to avoid any unauthorized modifications. The payload portion could also be made tamper proof.

[10] Customization of the content group program is contained within the payload portion. This embodiment includes XML data in the payload portion to somehow indicate the group of objects for download, among other things. Other embodiments could use other techniques to store information in the payload portion. In one embodiment, the payload portion includes the transaction session identifier, a user identifier, a number of URIs for the objects, a description for each object, and configuration information. The operator of the origin server uses the configuration information to affect how the content group program behaves. For example, the operator may limit download bandwidth, may specify how many objects should be simultaneously downloaded, may specify alternative paths for the objects, specify display properties, specify DRM limitations, etc. Instead of including information in the payload portion, the payload could specify the transaction session identifier and a remote database could retrieve any other information associated with that transaction session identifier.

[11] Building the content group program is not resource intensive. The program portion along with any configuration information is retrieved for the operator. The remaining XML data is compiled for the transaction session. A concatenation is performed for the program and payload portions. This may include compiling the concatenated portions or not. The file name for the content group program is formulated. In this embodiment, a date code is used to create the file name. The file name could include other information also.

[12] In one embodiment, there is no payload portion. The content group program is identical, but the file name is customized with a token that uniquely identifies the transaction session. Execution of the content group program forwards the token to the origin server operator for retrieval of the other information used to download the objects associated with the content group program. Some embodiments could use tool tip information to store the token. Further, the tool tip information could store data on the objects to help identify what the content group program corresponds to.

[13] While the principles of the invention have been described above in connection with specific apparatuses and methods, it is to be clearly understood that this description is made only by way of example and not as limitation on the scope of the invention.